DEPARTAMENTO
DE COMPUTACION
Facultad de Ciencias Exactas y Naturales - UBA

Trabajo Práctico 2

Rutas en internet

21 de octubre de 2014 — Teoría de las Comunicaciones

Integrante	LU	Correo electrónico
Barbeito, Nicolás	147/10	nicolasbarbeiton@gmail.com
Garassino, Agustín Javier	394/12	ajgarassino@gmail.com
Vileriño, Silvio	106/12	svilerino@gmail.com

Instancia	Docente	Nota
Primera entrega		
Segunda entrega		

Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires
Ciudad Universitaria - (Pabellón I/Planta Baja)
Intendente Güiraldes 2160 - C1428EGA
Ciudad Autónoma de Buenos Aires - Rep. Argentina
Tel/Fax: (54 11) 4576-3359
http://www.fcen.uba.ar

Índice

1. Introducción

Los objetivos de este trabajo practico son varios, para la primera etapa de desarrollo del trabajo, realizamos una herramienta que realiza un traceroute utilizando el protocolo `ICMP` con `TTL` incrementales, utilizando scapy en python. Luego de tener la primera version de la herramienta, se le agregaron funcionalidades, entre ellas el calculo del RTT entre hops, un analisis estadistico de estos datos, incluyendo el calculo de promedio, desv. estandar y `zscore`. Además, dado que la segunda etapa del trabajo implica el analisis de las trazas a diferentes continentes y el descubrimiento de nodos distinguidos, particularmente enlaces submarinos, se utilizaron varias APIs públicas para la obtencion del nombre del host del hop a partir de su ip(`DNS Reverse Lookup`), asi como tambien la `geolocalizacion` de la IP, tanto en terminos de pais y ciudad, como tambien en terminos de latitud y longuitud en un planisferio. Para complementar los analisis, se realizaron graficos, de distribucion de las mediciones estadisticas y un grafico sobre un planisferio que muestra la traza por los diferentes hops distribuidos en el mundo.

2. Desarrollo

2.1. Implementacion de traceroute

En esta seccion explicaremos como fue realizada la herramienta en python que realiza el `traceroute`.

2.1.1. Envio de paquetes con ttl incremental

Utilizamos el protocolo `ICMP` y la tecnica de `TTL incrementales` para obtener los hops intermedios de origen a destino, esta tecnica consiste en enviar paquetes `ICMP` con ttl en un rango creciente, comenzando en 1, y al recibir una respuesta, dependiendo el tipo de respuesta ir armando la ruta, particularmente si la respuesta es **Time Exceeded**, quiere decir que el ttl del paquete se agoto en camino y obtenemos un hop intermedio de la direccion origen del paquete de la respuesta de `ICMP`. Si la respuesta es **Echo Reply**, quiere decir que el host destino fue alcanzado. El incremento del `TTL` continua en cada iteracion hasta o bien llegar al host destino, o bien llegar a un limite de TTL, usualmente 30 saltos en el traceroute de linux, limite que adoptamos para nuestra herramienta.

2.1.2. Control de hops sin respuesta - Timeout

Puede ocurrir que para cierto `TTL` el hop correspondiente no conteste, es por esto que agregamos un `timeout` asociado a la petición. Luego de este tiempo, el intento con dicho `TTL` es descartado, marcado el hop como desconocido y se procede a incrementar el `TTL` y continuar con el ciclo de la traza.

2.1.3. Medicion del RTT

Scapy nos provee de ciertos campos temporales en los paquetes, utilizamos estos campos para determinar el tiempo `Round trip time` o `Tiempo de ida y vuelta`, en los paquetes de envio y respuesta de la peticion, hay campos indicando el `Unix Time` en el que fueron enviados y recibidos, correspondientemente, al realizar la resta entre ellos, obtenemos la medicion que buscabamos y al multiplicarla por 1000 obtenemos el tiempo en milisegundos.

2.2. Informacion obtenida de hops

Para refinar el analisis, como mencionamos en la introduccion, se utilizaron APIs para obtener `metadatos` acerca de los hops con IP descubierta.

2.2.1. Reverse DNS Lookup

Utilizamos el `Servicio Web` situado en `http://api.statdns.com/x/<IpAddress>` para resolver los nombres de host de los hop a partir de su direccion IP.

2.2.2. IP Geolocalization Lookup

Utilizamos el `Servicio Web` situado en `http://freegeoip.net/json/<IpAddress>` para obtener datos de posicion geografica de los hops utilizando su direccion IP.

2.3. Problemas surgidos durante el desarrollo

2.3.1. Hops con RTT_i negativo

Si pensamos a internet como un grafo y los hosts origen y destino como 2 nodos, la traza es el camino entre origen y destino en el grafo, si pensamos que el `TTL` restringe la cantidad de saltos entre nodos consecutivos que pueden darse a partir del host origen, uno esperaria que el `RTT` fuera

creciente, dado que es acumulativo. Pero al haber varias rutas posibles, diferente congestion a cada instante, diferentes tiempos de encolamiento a cada instante, y otras variables, es posible que en cada pedido el camino no sea el mismo. Esto produce que la medicion de `RTT` incrementando los `TTL` no sea creciente.

Al momento de calcular los $RTT_i = RTT_{(acum,i)} - RTT_{(acum,i-1)}$ de cada hop la situacion explicada arriba puede producir `RTT incrementales` negativos. Por lo general los nodos distinguidos **tienen una variacion importante de RTT acumulado y no son afectados usualmente por este problema**, de forma que no implementamos ninguna solución a este problema porque no nos afecta en los resultados que buscamos.

2.3.2. Hops sin un hop inmediato anterior valido

Otro problema surge al calcular $RTT_i = RTT_{(acum,i)} - RTT_{(acum,i-1)}$, el hop $i-1$ puede no existir, o ser desconocido. En el primer caso, coincide que $RTT_1 = RTT_{(acum,1)}$ asi que no hay mayores inconvenientes. En caso de ser desconocido el hop anterior, asumiendo que el primer hop, usualmente el gateway del host origen es siempre descubierto, el enfoque utilizado para solucionar este problema fue iterar hacia atras desde el i-esimo hop, y realizar el calculo $RTT_i = RTT_{(acum,i)} - RTT_{(acum,j)}$, donde j es el indice del primer hop detrás del i-esimo hop. El problema que trae esto, es que agrupamos hops desconocidos como uno solo, lo que nos va a afectar el descubrimiento de hops distinguidos, dando falsos positivos. Una segunda iteracion sobre este enfoque fue simular un RTT equitativo entre todos los hops intermedios ocultos entre el i-esimo y el j-esimo hop y asignarle $RTT_i = \frac{RTT_{(acum,i)} - RTT_{(acum,j)}}{cant.hopssalteados}$. a cada hop intermedio entre los hops j e i. Finalmente utilizamos esta segunda iteracion de la solucion para nuestro analisis.

2.3.3. Varias iteraciones para descubrir rutas y rtt promedio

Dado que usualmente no cambian de forma significativa las rutas ni los tiempos para corridas muy inmediatas entre ellas, decidimos no analizar esto realizando varias iteraciones programaticamente.

2.3.4. TCPTraceroute para mejorar descubrimiento de hops

Para intentar disminuir la cantidad de nodos desconocidos, realizamos una prueba enviando paquetes TCP y UDP adaptando las condiciones de terminacion del algoritmo de traza, los resultados obtenidos fueron identicos a las pruebas realizadas con ICMP con lo cual no avanzamos por este camino.

2.4. Estadisticas y Nodos distinguidos

Para la deteccion de nodos distinguidos en la ruta calculamos una serie de estimadores estadisticos.

2.4.1. Promedio y Desv. Estandar del RTT_i

Al final de la traza, se calcula el promedio y la desviacion estandar del $RTTincremental$ de cada hop en la traza.

2.4.2. Calculo de ZScore para cada hop

Realizando el calculo $Zrtt_i = \frac{RTT_i - RTT_{prom}}{RTT_{stdev}}$ asignamos un puntaje signado a cada hop, si $Zrtt_i$ es negativo, se encuentra por debajo de la media, si es positivo, se encuentra por encima de la media.

2.4.3. Eleccion empírica del umbral para deteccion de nodos distinguidos

Tomamos un umbral arbitrario $\lambda = \frac{1}{2}$, todos los hops que se encuentren con puntaje por encima de este umbral, son considerados distinguidos. Dado que obtuvimos buenos resultados con este umbral, no vimos la necesidad de ajustarlo.

2.5. Gráficos y análisis realizados

Para los experimentos realizados a ciertos hosts de distintos continentes se presenta la informacion con las siguientes herramientas.

- `Tabla de hops de la traza:` Se muestra una tabla informando los hops entre origen y destino, con el `RTT acumulado`, el `RTT incremental` y el `zscore` de cada hop, y donde es posible, la resolucion del `host-name` y la `geolocalizacion` del host.
- `Distribucion de` RTT_i`:` Se muestra un grafico de barras donde el eje x indica las IP de los hops y en el eje Y de muestra una barra indicando el `RTT incremental` entre dicho hop y el anterior.
- `Distribucion de` RTT_{ttl} `acumulado:` En este grafico mostramos de izquierda a derecha, los hops en orden desde origen a destino y en el eje Y el `RTT acumulado` desde origen hasta este hop.
- `Distribucion de` $ZScore_i$`:` Para cada hop en el eje x, en este grafico, mostramos en el eje Y una barra indicando el puntaje estandar otorgado a este hop.

2.5.1. Traza sobre el planisferio

Con el fin de mostrar de forma clara los datos geograficos recolectados con la API mencionada en secciones anteriores, utilizamos una libreria de python que grafica un punto rojo de tamaño variable segun el score asignado y un arco verde entre los hops. Esperamos que los nodos submarinos obtenidos de forma estadistica utilizando el puntaje estandar correspondan con los arcos entre los enlaces submarinos.

3. Experimentacion

3.1. www.unsw.edu

3.2. www.cam.ac

3.3. home.web.cern.ch

4. Experimentacion Adicional

De ciertas experimentaciones adicionales realizadas por curiosidad pero no presentadas en este informe dado el alcance del trabajo obtuvimos resultados extraños, como servidores de ISP intermedios contestando `Echo Reply` como si fueran el host destino, creemos que esto se debe a un cacheo intermedio que realiza el ISP para minimizar la redireccion de trafico fuera del pais. Otros experimentos realizados sobre la red movil 3g anclando la conexion a una pc, mostraron una cantidad de saltos muy grande dentro del pais, con tiempos de `Round Trip Time` muy altos en comparacion al enlace submarino hasta el host destino, los nodos distinguidos de la ruta quedaron enmarcados dentro del pais, lo cual podria explicarse por el nivel de congestion de la red de telefonia de la capital. Otros experimentos realizados sobre redes 3G arrojan una cantidad innecesaria de hops entre continentes para llegar a destino, particularmente los paquetes iban de Argentina, a Uruguay, luego a Italia, luego a Estados Unidos, volvian al Reino Unido, y finalmente a destino en europa.
Algunos de los datos de los experimentos mencionados pueden encontrarse en el comprimido adjunto por fuera del informe.

5. Conclusiones

Como conclusion del trabajo pudimos entender como funciona el protocolo ICMP para realizar una traza con ttl incrementales, y obtener una buena aproximacion de los enlaces submarinos utilizando las herramientas desarrolladas en el trabajo. Los resultados fueron variados, en la seccion de `Experimentos Adicionales` se mencionan algunos resultados curiosos. En general es interesante la cantidad de datos, nombres de hosts y geograficos que pueden obtenerse a partir de la realizacion de una traza. Tambien son interesantes las aplicaciones practicas de las trazas, los datos obtenidos pueden utilizarse para diagnostico, por ejemplo se quiere saber en que momento se pierde un paquete, con la traza puede saberse en que hop se pierde. Otra posible aplicacion podría ser para aumentar la performance en el diseño de agregacion de enlaces adicionales a internet, alivianando la carga en los hops con mucho `RTT incremental`.